Bits & Bytes

INTERNATIONAL SOFTWARE CONSIDERATIONS: NTSC vs. PAL Systems

— ATARI CONFIDENTIAL —

A Warner Communications Company

Reproduction is forbidden without the specific written permission of ATARI, INC., Sunnyvale, CA 94086. No right to reproduce this document, nor the subject matter thereof, is granted unless by written agreement with, or written permission from the Corporation.

Every effort has been made to ensure that this text accurately documents this product of the ATARI Computer Division. However, because of the ongoing improvement and update of the computer software and hardware, ATARI, INC. cannot guarantee the accuracy of printed material after the date of publication and cannot accept responsibility for errors or omissions.

COPYRIGHT 1982 ATARI, INC.

PAM—04/23/82
BB001-A

Essentially, the problem encountered in developing compatibility in international software revolves around the synchronization of the computer hardware to a television signal. In the United States, the standard is 60 frames per second: NTSC system. Some European countries use a PAL standard: 50 frames per second. Therefore, in conforming software internationally, problems develop in the areas of timing and screen image. This note intends to provide helpful information pertaining to the development of software for a potential international and domestic market.

Assuming that development is on the NTSC system with a software check for PAL, perform the following steps for international operation.

TIMING TROUBLES

STEP 1. Determine if the system is NTSC or PAL.

PEEK at location 53268. The value contained in this register on an NTSC system is 15. Should another value occur, adjustments must be made to any software timers used in real-time program applications.

Example of PAL system check.

```
IF PEEK(53268) = 15 THEN SYSTEM = NTSC
```

STEP 2. Determine application use of RTCLOK.

RTCLOK is a three-byte value, residing at decimal locations 18, 19, and 20. Within RTCLOK, the highest address is the least significant byte; the lowest address is the most significant byte. RTCLOK increments by 1 at intervals of 1/60 of a second on an NTSC system, 1/50 of a second on PAL.

Use of the RTCLOK in application programs can vary between its function as a "counter" or a "timer"; the program can either delay by monitoring for a specific time lapse or waiting for a predefined value.

STEP 3. Adjust RTCLOK.

RTCLOK pulses faster at 60 Hz than 50 Hz. To have the same "effective" time, adjust the value by a 5/6 multiplication factor.

To arrive at the same "effective" value requires a 5/6 adjustment.

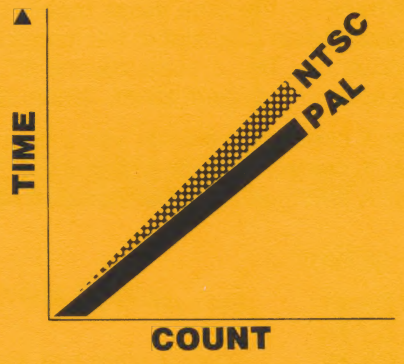

Figure 1 Elapsed Time Comparison

When the application calls for taking two RTCLOK values and subtracting to determine a time interval, adjust the value by a 6/5 multiplication factor.

To arrive at the same "effective" time requires a 6/5 adjustment.

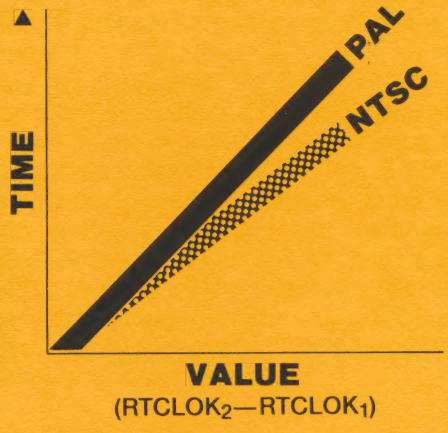

Figure 2 Value Comparison

STEP 4. Adjust system timers 1-5. (See Table 1.)

Essentially, the procedure for using a system timer requires loading its registers X, Y with a specific value. The OS automatically decrements this value during VBLANK. Whenever the time value reaches a 0, the OS will either send an interrupt to the microprocessor or set a status bit. The system routine SETVBV initializes the specified system timer. Within SETVBV,

- Register X contains the timer value high byte
- Register Y contains the timer value low byte
- Register A contains the timer number 1-5

Example of SETVBV.

```
LDA #1      ;Set system timer 1
LDY #0
LDX #2      ;value is $200 VBLANK
              periods
JSR SETVBV;Call system routine to set
              timer
```

To simulate NTSC operation on a PAL system, do the appropriate 5/6 multiplication computation on the X,Y registers before the call to JSR.

DESIGN CONSIDERATION

When game play that is developed on a 60 Hz refresh rate runs on a 50 Hz system, a noticeable sluggishness occurs; react time is significantly impaired. One way to regain the speed of play is to effect an approximation of the 60 Hz refresh rate on the PAL. Conceptually, the idea is to imitate faster movement. For example, consider a horizontal movement of one pixel per frame on NTSC. To simulate that movement on PAL, move the object one pixel per frame for four frames; then on the fifth frame, move the object two pixels.

When the NTSC system was adapted for PAL, an adjustment was made in the OS chips to compensate for the difference between a 50 Hz and a 60 Hz system. This compensation results in the ability to generate tapes on an NTSC system and have those same tapes load into the PAL. However, the problem recurs in the critical area involving those tapes with both audio and digital tracks, where the timing of sync marks is crucial. Keep in mind that extensive use of system timers will aggravate the problem. Check for the use of any system timers and follow the instructions given.

TABLE 1
SYSTEM SOFTWARE TIMERS

Within the OS are six system software timers:

Name	Location	Vector or Flag
RTCLOK	[$0012,3]	none
CDTMV1	[$0218,2]	CDTMA1 [$0226,2]
CDTMV2	[$021A,2]	CDTMA2 [$0228,2]
CDTMV3	[$021C,2]	CDTMF3 [$022A,1]
CDTMV4	[$021E,2]	CDTMF4 [$022C,1]
CDTMV5	[$0220,2]	CDTMF5 [$022E,1]

These timers perform real-time counts or use VBLANK for period counts and are affected by refresh rates.

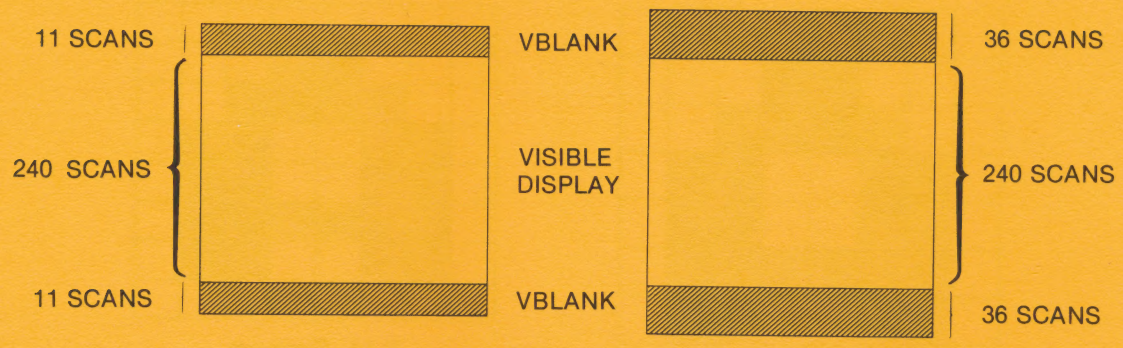

Figure 3 NTSC and PAL Screen Image Display

SCREEN DISPLAY DIFFERENCES

In a video sense, a **field** consists of a set of odd or even scan lines, and a **frame** is an "interlaced" composite of both fields. In computer adaptation to the television beam, interlace is ignored and the frame is considered to be the total number of scan lines in a field.

Operating systems on the PAL and NTSC work with no modification to screen graphics. Both systems display 240 visible lines with the remainder of the scan lines being VBLANK. Total scan lines on NTSC is 262; PAL has a total of 312. Taking into account the condition of overscan, the average television set is tweaked to prevent seeing VBLANK. By using 24 scan lines as background, 192 scan lines are dedicated to playfield. In adapting NTSC to PAL, 50 additional scan lines are in the VBLANK area. Therefore, there are no modifications to software that will eliminate the appearance of the VBLANK lines on the PAL system.

Note: When the NTSC system was adjusted for PAL, a compensation was made so that a display list has compatibility. Even though the additional VBLANK lines are split symmetrically upon the screen, the OS begins the display of screen data so that the program does not have to take the scan line count into consideration.

Overscan is a significant problem in adaptation to PAL. Literally, close to 38% of the screen image is dedicated to background and VBLANK areas. Bear in mind that although there are software-oriented solutions, such as rewrite of display lists if conditions are checked and indicate a PAL system, any optimization for PAL automatically effects a de-optimization for NTSC. Also, screen image corrections and special condition software carry memory-expensive overhead.

Because the PAL system displays a total of 50 additional scan lines in the same allocated space as NTSC, mathematically, the PAL has a 19% denser screen. Visually, the objects appear to be 4/5 as **tall.** Although there is a significant vertical change, horizontal densities between the two screens are within 2% of each other. It follows that the image takes on a squat characteristic when displayed on the PAL system; for example circles on the NTSC translate into ellipses on PAL.

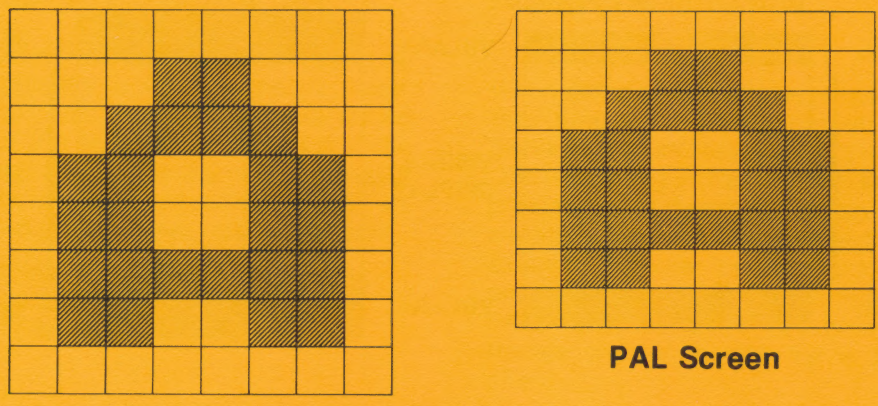

Figure 4 Display Representation of NTSC vs. PAL

Because the character set is defined on the 8x8 grid pattern and due to the limitations imposed by the mechanics of the display screen, text mode characters are **fixed**. Software corrections to text mode characters cannot be effected.

Graphics mode displays, however, can be reshaped. To simulate the same circle on both the NTSC and PAL systems requires that the graphics be expanded vertically by a 5/4 multiplication factor.

COLOR CONSIDERATIONS

PAL systems are equipped exclusively with GTIA. Color generation for the PAL/GTIA is a different technique than the NTSC/GTIA. Because the hardware doesn't have the right resolution, colors are approximate with slight variance and a different spectrum.

Note: If the method of artifacting is used, all selected colors will appear reversed from the CTIA to GTIA chips.